

Appl. No. 10/815,294  
Amtd. dated August 21, 2006  
Reply to Office Action of April 19, 2006

Amendments to the Drawings:

Included in the amendment are an "Annotated Sheet Showing Changes" and a "Replacement Sheet" for Fig. 2C. In Fig. 2C, a duplicate reference number 210 was inadvertently used. As described in the text on page 9, line 7, "Fig. 2C illustrates a programmer's view 220". Fig. 2C has been amended changing the reference number 210 to 220.

REST AVAILABLE COPY

Appl. No. 10/815,294  
Amdt. dated August 21, 2006  
Reply to Office Action of April 19, 2006

### Remarks

The present amendment responds to the Official Action dated April 19, 2006. A petition for a one month extension of time to respond and authorization to charge Deposit Account No. 50-1058 the large entity extension fee of \$120 accompany this amendment. The Official Action rejected claim 1 under 35 U.S.C. §102(b) as being anticipated by Saulsbury et al. U.S. Patent Publication No. 2002/0032710 (Saulsbury). Claim 2 was rejected under 35 U.S.C. §103(a) based on Saulsbury. Claims 3-10 and 16-19 were rejected under 35 U.S.C. §103(a) based on Saulsbury as applied to claim 2 in view of Nair et al. U.S. Patent No. 6,944,747 (Nair). Claims 11-15 were rejected under 35 U.S.C. §103(a) based on Saulsbury in view of Nair and further in view of Dowling U.S. Patent No. 6,823,505 (Dowling). These grounds of rejection are addressed below.

Claims 3, 4, 11, 15, and 19 have been amended to be more clear and distinct. Claims 1-19 are presently pending.

### Amendment to the Specification

The paragraph beginning at page 21, line 10 has been amended to correct an inadvertent use of an incorrect name "storage unit" for block 835. Block 835 should be named storage device 835 consistent with the usage at page 22, in lines 8, 10, and 17.

### The Art Rejections

As addressed in greater detail below, Saulsbury, Nair, and Dowling do not support the Official Action's reading of them and the rejections based thereupon should be reconsidered and

Appl. No. 10/815,294  
Amdt. dated August 21, 2006  
Reply to Office Action of April 19, 2006

withdrawn. Further, the Applicant does not acquiesce in the analysis of Saulsbury, Nair, and Dowling made by the Official Action and respectfully traverses the Official Action's analysis underlying its rejections.

The Official Action rejected claim 1 under 35 U.S.C. §102(b) as being anticipated by Saulsbury. Saulsbury describes a processor having transpose sub-instructions and hardware associated with the transpose sub-instructions. A transpose sub-instruction, when executed, causes packed data values to be accessed at source register locations specified by the sub-instruction and then specific data elements are selected from the accessed packed data values. Once the specific data elements have been selected, they are rearranged in a packed data format prior to storing the rearranged data elements in a register at a destination register location specified by the sub-instruction. Saulsbury further indicates that two transpose sub-instructions may cooperate in execution to improve performance of transpose operations.

The Examiner states that Saulsbury describes a processor address translation apparatus for translating an instruction operand address to a different operand address. Regardless of the correctness of this statement, Saulsbury does not describe "translating an instruction operand address to a different operand address" as presently claimed in claim 1. Rather, when executing two transpose sub-instructions, Saulsbury multiplexes bit-fields from four addressed source registers to generate two reordered sets of packed data values that are then stored in registers at destination addresses specified in the transpose sub-instructions. This approach is specifically shown in Saulsbury Fig. 5 in conjunction with Fig. 6A. For example, one of the four source register RW 508-1 contains four data values a, b, c, and d, and another source register RX 508-2

Appl. No. 10/815,294  
Amdt. dated August 21, 2006  
Reply to Office Action of April 19, 2006

contains four data values e, f, g, and h. The four destination values are also shown in Fig. 5, with, for example, RP 504-1 containing destination values a, e, i, and m, and RQ 504-2 containing destination values b, f, j, and n. The four registers 508 are shown in matrix 500 before performing the transpose operation and the four destination registers 504 are shown in transposed matrix 502 after performing two transpose operations. In Saulsbury's Fig. 6A Step 1, two instructions, trans0 RW, RX, RP 600 and trans0 RY, RZ, RR 604, are shown. Trans0 RW, RX, RP 600 specifies the source operand addresses RW and RX which are not translated and causes the reading of source registers RW and RX. The first source address (Rs1) W, as shown in Fig. 6A, has data values a, b, c, and d that are the same data values as the contents of the register RW 508-1 prior to the transpose operation as shown in Fig. 5 matrix 500. In a similar manner, the second source address (Rs2) X, as shown in Fig. 6A, has the values of e, f, g, and h that are the same data values as the contents of the register RX 508-2 prior to the transpose operation as shown in Fig. 5 matrix 500. The trans0 RY, RZ, RR 604 instruction also reads the specified source address registers RY and RZ without translation. The two trans0 instructions 600 and 604, when executed, cause the selection of specific data elements from the accessed source registers' packed data values. The selected data elements are then rearranged to form new packed data values. The rearranged new packed data values are then loaded in registers at the destination register addresses specified in the two trans0 instructions. Saulsbury Figs. 5 and 6A and paragraphs 53-55. The destination register addresses **are not translated**. Saulsbury does not teach and does not make obvious an apparatus or method as presently claimed which addresses the problem of translating an instruction operand address to a different operand

Appl. No. 10/815,294  
Amdt. dated August 21, 2006  
Reply to Office Action of April 19, 2006

address. The present claims are not taught, are not inherent, and are not obvious in light of Saulsbury.

Claim 2 was rejected under 35 U.S.C. §103(a) based on Saulsbury. The Examiner cites paragraph 67 on page 6 of Saulsbury as purported to describe "a plurality of translation parameters and address translation functions supporting a plurality of translation patterns" as claimed in claim 2. The table associated with paragraph 67, illustrates the type of VLIW processor, the width of the data elements, and the width of the registers that are compatible to obtain the "sixteen element matrix" associated with each variation in the table. "The sole table indicates some of the possible variations of this invention for performing a transpose in one issue." Saulsbury, paragraph 67. The table does not describe "a plurality of translation parameters and address translation functions supporting a plurality of translation patterns" as presently claimed in claim 2 since the table presents only variations in processor and data organizations to support a single transpose operation. Claim 2 is not taught, is not inherent, and is not obvious in light of Saulsbury.

Claims 3-10 and 16-19 were rejected under 35 U.S.C. §103(a) based on Saulsbury as applied to claim 2 and in view of Nair. Nair addresses "an indexing methodology 300" for defining the resolution of supported data elements, such as 4-bit, 8-bit, 16-bit, and so forth, a starting index of a first element of a first source matrix, a starting index for a first element of a second source matrix, a starting index for a first element of a destination matrix, and an interval between elements for selecting among successive elements of the matrices. Nair, col. 8, lines 47-50, col. 10, line 65 – col. 11, line 27.

Appl. No. 10/815,294  
Amtd. dated August 21, 2006  
Reply to Office Action of April 19, 2006

Regarding claim 3, the Examiner indicates that Nair describes an apparatus for matrix processing that can manipulate address bits using matrix multiply, addition, subtractions, bit-reverse operations etc. and cites Nair column 12, table 1 for support. Nair, column 12, table 1, is a listing of specific matrix operations that can be performed on the data elements of the matrix and does not describe operations that can be performed on an "operand address input" as presently claimed in amended claim 3. Nair, col. 12, lines 14-22.

Turning to claim 4, this claim defines the "instruction encoded with an operand address" of claim 1 as being "a block load instruction". The Examiner cites Saulsbury's paragraph 66 as describing "the matrix operates on 16 elements in a 4 by 4 matrix". To the contrary, Saulsbury's paragraph 66 describes the operation of a trans1 sub-instruction which causes bit fields 512-10, 512-12, 512-14, and 512-16 to be selected and written to the destination registers 504-2 and 504-4, as shown for example in Saulsbury's Fig. 8. Saulsbury's paragraph 66 has no apparent relevance to a block load instruction as claimed in claim 4.

Regarding claim 7, the Examiner incorporates the rationale of the rejection of claims 1 and 2. As noted above, the rationale for the rejection of claims 1 and 2 is faulty. Further, Saulsbury's Fig. 3 does not illustrate "an instruction encoded with an operand address and control information indicating the operand address is to be translated as part of the instruction's execution" as claimed in claim 7, but rather indicates data types as stated in the descriptive labels under each data type depicted in Fig. 3. Also, Saulsbury's paragraphs 45, 46, 51, and 52 do not describe a register file indexing address translation apparatus. Saulsbury's paragraph 45 describes a matrix transpose instruction format. Saulsbury's paragraph 46 describes a limitation

Appl. No. 10/815,294  
Amdt. dated August 21, 2006  
Reply to Office Action of April 19, 2006

that two transpose sub-instructions are issued at a time and the processing paths associated with both sub-instructions have access to the register files associated with the processing paths. Paragraph 51 describes a data organization matrix 500 as stored in a register file prior to performing the transpose operation and a transposed matrix 502 data organization as stored in a register file after performing the transpose operation. Paragraph 52 suggests that larger matrices may be broken into "four-by-four chunks and manipulated separately". None of the cited paragraphs, 45, 46, 51 or 52, describe, for example, a "register file indexing (RFI) address translation apparatus" where a "sequence of RFI operand addresses" are generated by an "RFI update unit" and the "sequence of RFI operand addresses" are translated "to form a sequence of different operand addresses in accordance with the translation pattern" as presently claimed in claim 7.

Clarification is requested from the Examiner as to how independent claim 16 is rejected based on the same rationale as in the rejection of the amended dependent claim 4. Claim 4 claims "the instruction is a block load instruction" and claim 16 claims a "processor address translation method". The rationale for rejecting claim 4 is not supported as addressed in detail above. If the rejection of claim 16 is truly based on the rationale for rejecting claim 4 it should be withdrawn on the same basis. However, claim 16 distinguishes from the references in same manner as claim 1 placing it in order for allowance.

Claim 18 was rejected based on the same rationale as in the rejection of claims 1 and 3. As described above, that analysis is not supported by Saulsbury.

Appl. No. 10/815,294  
Amtd. dated August 21, 2006  
Reply to Office Action of April 19, 2006

The Official Action rejected claims 11-15 under 35 U.S.C. §103(a) based on Saulsbury in view of Nair and further in view of Dowling. As presently claimed, the "address translation memory device" of amended claim 11 comprises "a first read address input", "a storage device having data accessible at addressable locations, a second read address input internal to the address translation memory device for selecting data from the storage device during read operations, and a data output port", and "an address translation unit ... the address translation unit translating the first read address input in accordance with the translation pattern, to the storage device second read address input for reading data from the storage device at a translated address during a read operation". See, for example, Fig. 8A and page 22, lines 7-8 of the present invention.

Dowling Fig. 2 provides a programmable address arithmetic unit (AAU) 212 that augments a fixed function AAU 106 to generate different addressing modes when accessing data from a separate data memory 120. Dowling generates different addresses based on address values that are stored in data memory address register set 102. Dowling's data memory 120 is clearly separate from the programmable AAU 212 having two multiplexers 122 and 203 and the address register set 102 between the programmable AAU 212 and the data memory 120. Further, the processor architecture 200 of Fig. 2 is shown with independent program bus 101 connected to a program memory 107 and an independent data bus 112 connected to data memory 120 which is typical of a "Harvard-type architecture" that uses a data memory separate from the program memory. Both memories are also shown as standard type memories having a single address input and an instruction/data output. Dowling, Fig. 2, col. 7, lines 48-59 and col. 8, lines 5-9 and

Appl. No. 10/815,294  
Amdt. dated August 21, 2006  
Reply to Office Action of April 19, 2006

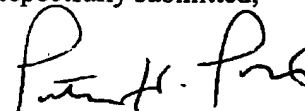
14-16. Saulsbury, Nair, and Dowling do not teach and do not make obvious an address translation memory device having an internal address translation unit and a storage device with a second read address input internal to the address translation memory device as presently claimed in amended claim 11. Nair and Dowling do not separately or collectively cure the deficiencies of Saulsbury addressed at length above.

Since dependent claims depend from and contain all the limitations of the independent claims, claims 2-6, 8-10, 12-15, 17, and 19 distinguish from the references in the same manner as claims 1, 7, 11, 16, and 18, respectively, and thus these claims are in order for allowance.

#### Conclusion

All of the presently pending claims, as amended, appearing to define over the applied references, withdrawal of the present rejection and prompt allowance are requested.

Respectfully submitted,



Peter H. Priest  
Reg. No. 30,210  
Priest & Goldstein, PLLC  
5015 Southpark Drive, Suite 230  
Durham, NC 27713-7736  
(919) 806-1600

BEST AVAILABLE COPY

SERIAL NO.: 10/815,294  
PETER H. PRIEST (919-806-1600)

*Annotated Sheet  
Showing Changes*

2/13

FIG. 2A

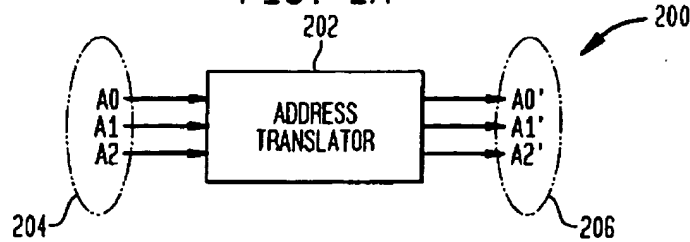


FIG. 2B

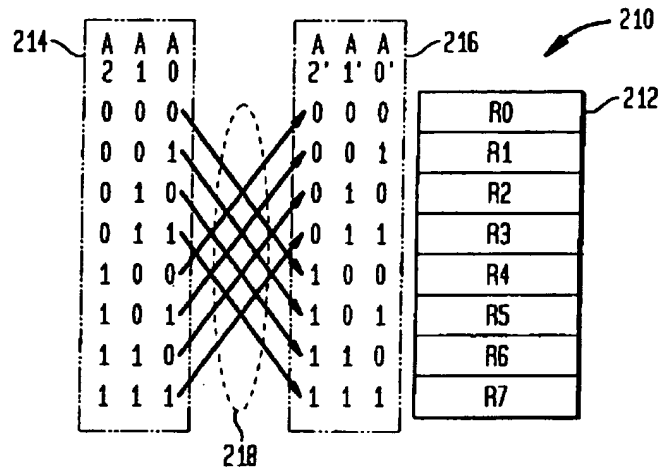
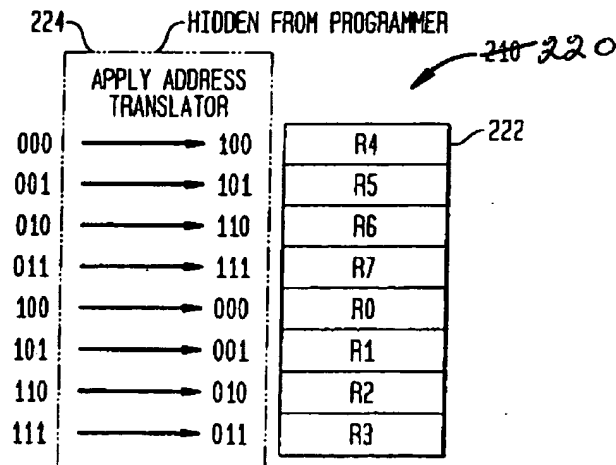


FIG. 2C



REST AVAILABLE COPY